

Sensor Temperatura - Termocupla Sumergible DS18B20

Es una sonda de temperatura sumergible que contiene el sensor de temperatura DS18B20 (Consultar el datasheet) el cual permite medir temperaturas hasta 125°C, además la termocupla esta sellada en un envoltorio que permite sumergirlo en líquido o protegerlo de la intemperie. Dado que es un sensor digital, la señal leída no se degrada debido a la distancia del cableado.



Figura 1. Sensor de temperatura sumergible DS18B20 (Fuente: Propia).

Se comunica utilizando el protocolo 1-Wire (One Wire), el cual es un protocolo de comunicación en serie basado en un bus con un maestro y varios esclavos de una sola línea de datos por la que se alimentan y se intercambian datos, utilizando por supuesto una referencia a tierra común de todos los dispositivos.

Esto conlleva a que el sensor solo utilizará 3 hilos hacia el microcontrolador o sistema embebido con una precisión de $\pm 0.5^{\circ}\text{C}$ y con una resolución de 12 bits.

También pueden utilizarse varios sensores sobre el mismo pin ya que internamente viene programado con un ID único de 64 bits para diferenciarlos, permitiendo realizar mediciones de temperatura en múltiples puntos.

El rango de funcionamiento es de 3 a 5V por lo que se puede utilizar en prácticamente cualquier sistema de que use microcontroladores.

Aplicaciones:

- Controles térmicos
- Sistemas industriales
- Termómetros

Características del cable:

- Tubo de acero inoxidable de 6mm de diámetro por 30mm de largo.
- Largo: 91cm.
- Diámetro: 4mm.
- Contiene un sensor de temperatura DS18B20.
- En función de la producción, los cables del sensor pueden variar pero seguirán según estas especificaciones:
 - Si tienes 4 cables: Rojo es Vcc (positivo), Negro es GND (negativo) y Blanco es el cable de datos. La malla es GND.
 - Si tienes 3 cables: Rojo es Vcc (positivo), Azul/Negro es GND (negativo) y Amarillo/Blanco es el cable de datos.

Características del sensor:

- Interface tipo 1-Wire
- Capacidad de medida desde -55°C hasta +125°C (-67°F -hasta +257°F)
- Precisión de $\pm 0.5^{\circ}\text{C}$ desde -10°C hasta +85°C
- No necesita componentes externos
- Longitud de cable de 1 Metro
- Voltaje de trabajo 3~5.5VDC Max.

Encapsulado:

Existen tres encapsulados: TO-92, SO y μSOP , ver figura 2.

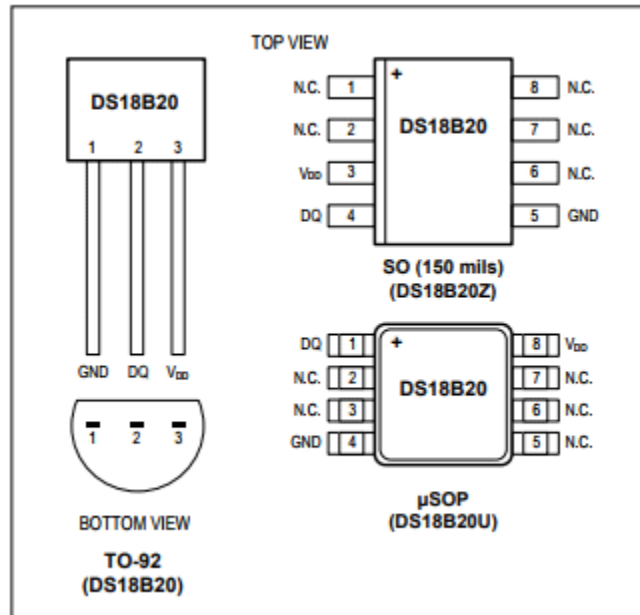


Figura 2. Tipos de encapsulados (Fuente: Datasheet [1])

Resolución del sensor de temperatura DS18B20:

Una de las características más interesantes de este sensor es que podemos trabajar con diferentes resoluciones. El DS18B20 admite resoluciones de 9-bit, 10-bit, 11-bit y 12-bit. Por defecto utiliza la resolución de 12-bit. Por supuesto, depende del tipo de resolución van a corresponder los incrementos en los valores de temperatura leído como se observa en la siguiente tabla:

Tabla 1. Incrementos en pasos de temperatura, según la resolución seleccionada en el sensor DS18B20. (Fuente: Datahseet [1]).

RESOLUCION	INCREMENTOS EN TEMPERATURA
9 bits	0,5°C
10bits	0,25°C
11bits	0,125°C
12bits	0,0625°C

La resolución dependerá de la precisión que se necesite en la aplicación en uso.

PROGRAMACIÓN

Primero se requieren las siguientes librerías, que se buscan en el gestor de librerías:

- OneWire para el protocolo de comunicación I2C
- DallasTemperature de Miles Burton

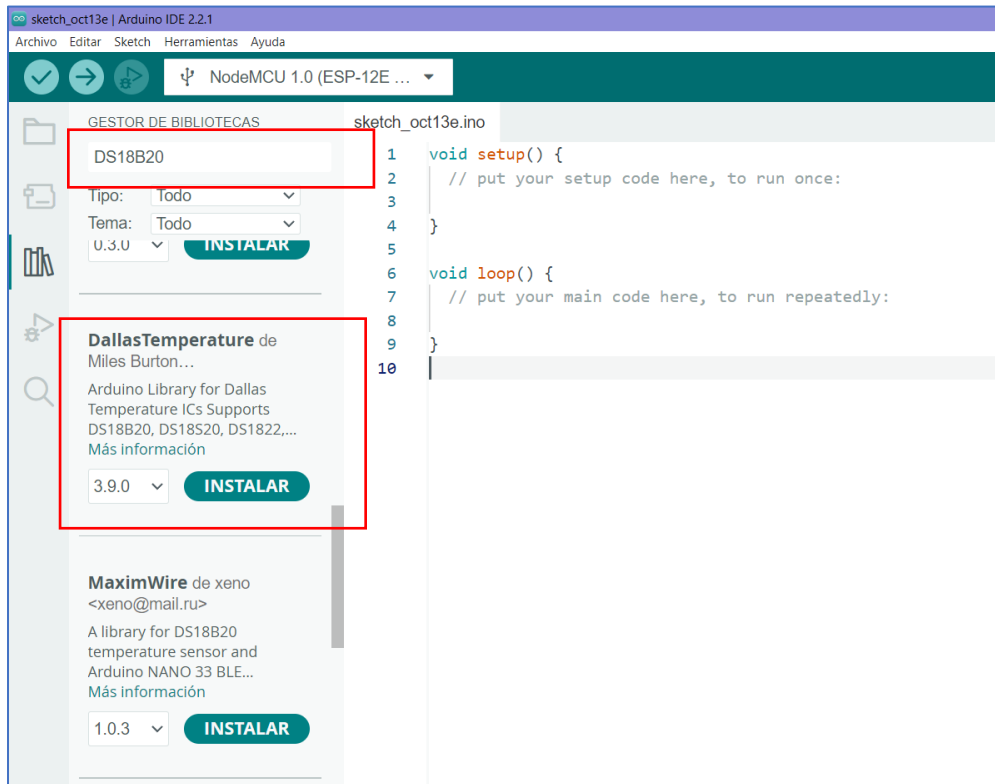


Figura 3. Utilizando el gestor de librerías “Dallas Temperature”.

En la nueva versión del entorno IDLE Arduino, una vez se desea instalar la librería DallasTemperature, él pide que se instale junto con las dependencias que requiere, ver figura 4, en este caso la dependencia es el protocolo de comunicación One Wire (1-Wire), ya que como habíamos mencionado anteriormente se necesita para comunicarse por una línea de datos.

Anteriormente, se requería instalar por separado las librerías.

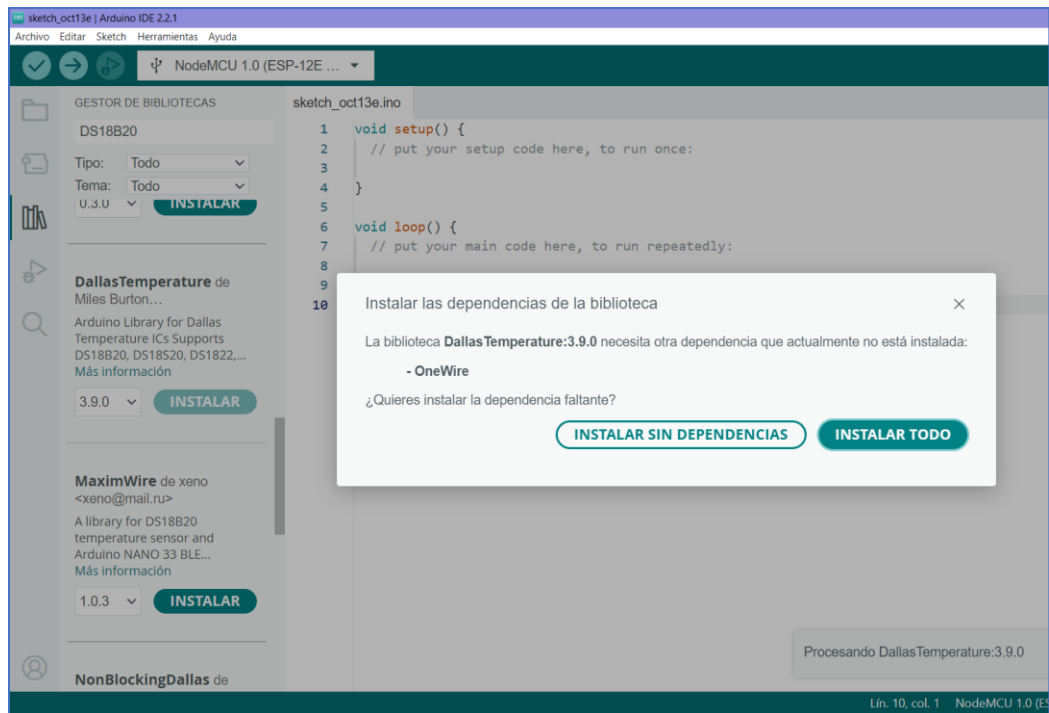


Figura 4. Solicitud para instalar la dependencia: Librería OneWire.

Una vez instaladas las librerías aparecerá en la salida del entorno IDLE de Arduino, ver figura 5.

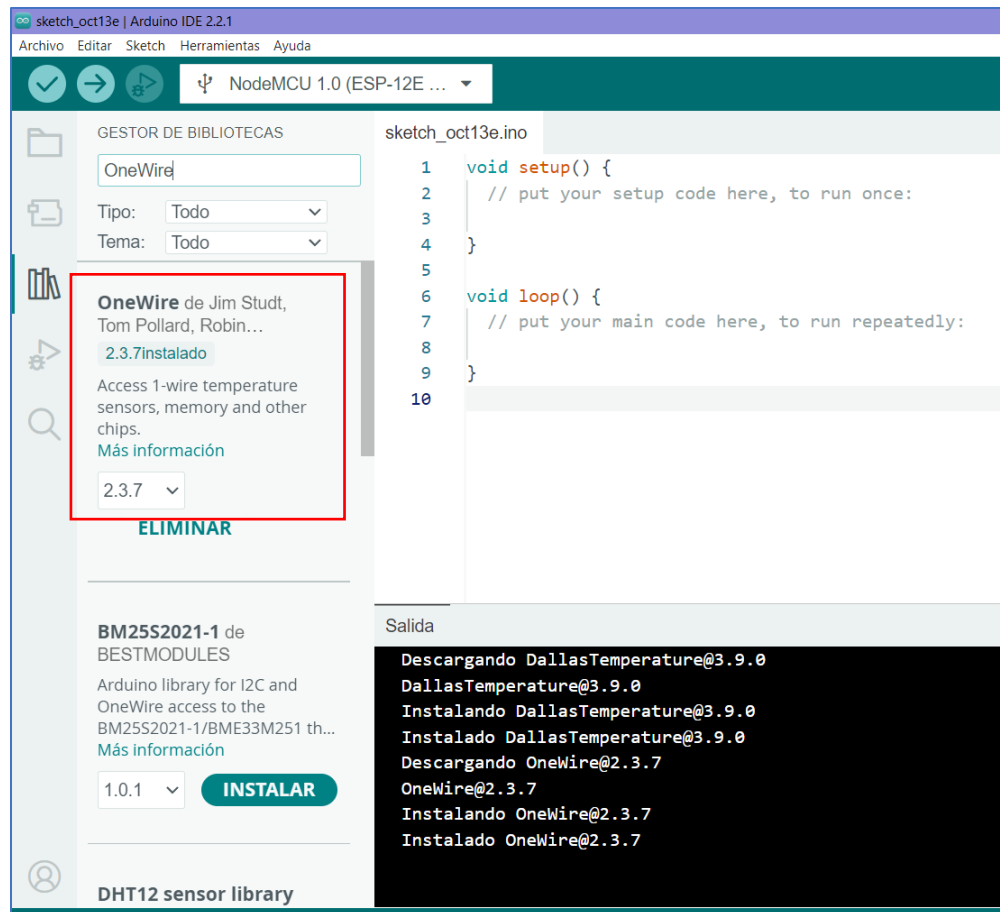


Figura 5. Instalación de las librerías: “DallasTemperature” y “OneWire”.

CÓDIGO PARA OBTENER LAS DIRECCIONES DE CADA SENSOR

Si en la aplicación se requiere utilizar varios sensores y no queremos utilizar varios pines digitales, porque seguramente los necesitaremos más adelante, se requiere identificar el sensor para ello utilizaremos el siguiente código:

```
//IDENTIFICACIÓN DE LA DIRECCIÓN DEL SENSOR DE TEMPERATURA, EN CASO
//DE QUERER UTILIZAR VARIOS SENSORES DSB18B20

#include <OneWire.h>
OneWire ourWire(2); //Se establece el pin 2 como bus OneWire
void setup(void) {
  Serial.begin(9600);
}
void loop(void) {
  byte addr[8];
  Serial.println("Obteniendo direcciones:");
  while (ourWire.search(addr))
```

```

{
  Serial.print("Address = ");
  for( int i = 0; i < 8; i++) {
    Serial.print(" 0x");
    Serial.print(addr[i], HEX);
  }
  Serial.println();
}
Serial.println();
ourWire.reset_search();
delay(2000);
}

```

CONEXIÓN DE LOS PINES DEL SENSOR DE TEMPERATURA DS18B20 CON EL ARDUINO

Debemos conocer que existen dos formas de alimentar el DS18B20. La primera sería la más simple y sería a través del pin V_{DD} . La segunda opción sería hacerlo a través del propio pin de datos DQ, en modo parásito.

En los dos modos posibles, siempre se pone una **resistencia pull-up** con el pin DQ. El motivo de esta resistencia es debido a la electrónica para controlar el bus de comunicación. Utiliza un FET de drenaje abierto que se comporta como una puerta AND.

Cuando todos los sensores conectados al bus 1-Wire no envíen datos, la línea de datos será igual a la tensión que suministremos (puede ser de 3V a 5,5V) debido a la resistencia pull-up. En el momento que un sensor empieza a transmitir, la línea cambia de estado y ya sabemos que hay un sensor transmitiendo datos.

Lo único que nos quedaría es determinar la **resistencia pull-up**. La resistencia que se vaya a utilizar dependerá de la longitud del cable. Por norma general utilizaremos siempre una de **4,7 k Ω** , les recuerdo esta información del cálculo de la resistencia pull-up a conectar nos la proporciona el datasheet del sensor.

En la siguiente tabla puedes ver la resistencia más adecuada dependiendo de la longitud del cable de datos.

Tabla 2. Resistencia PULL_Up, según la distancia del cable (m) (Fuente: Datahseet [1]).

RESISTENCIA PULL-UP	DISTANCIA EN METROS
4,7KOhmios	De 0m a 5m
3,3KOhmios	De 5m a 10m
2,2K Ohmios	De 10m a 20m
1,2K Ohmios	De 20m a 50m

Utilizando el modo de alimentación por el pin V_{DD} , la alimentación por este pin es desde 3V a 5,5V. No sobrepasar este límite o se dañará el sensor de temperatura DS18B20.

Realizando las siguientes conexiones, tomando como guía los pines de un DS18B20 con encapsulado TO-92:

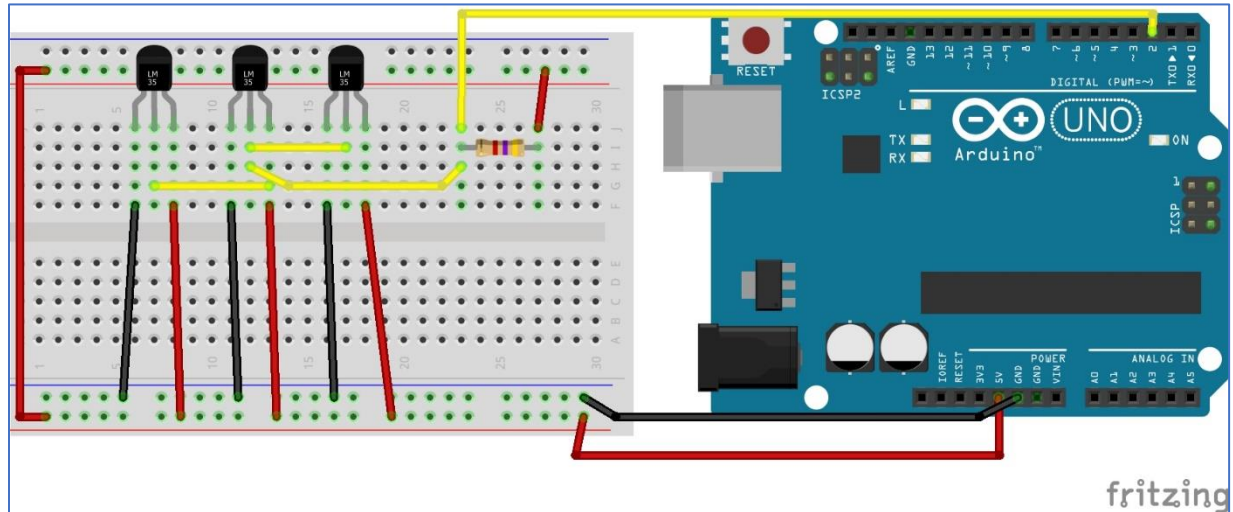


Figura 6. Conexiones del sensor DS18B20 al pin 2 de Arduino.

Se obtienen las direcciones o identificaciones propias de cada sensor ID:

```
Mensaje (Intro para mandar el mensaje de 'Arduino Uno' a 'COM4')
12:32:22.087 -> Obteniendo direcciones:
12:32:22.087 -> Address = 0x28 0x48 0xB5 0x7 0xD6 0x1 0x3C 0xC0
12:32:22.163 -> Address = 0x28 0x21 0xBD 0x7 0xD6 0x1 0x3C 0xAB
12:32:22.197 -> Address = 0x28 0xF5 0xC3 0x7 0xD6 0x1 0x3C 0x7C
12:32:22.230 ->
12:32:24.217 -> Obteniendo direcciones:
12:32:24.217 -> Address = 0x28 0x48 0xB5 0x7 0xD6 0x1 0x3C 0xC0
12:32:24.284 -> Address = 0x28 0x21 0xBD 0x7 0xD6 0x1 0x3C 0xAB
12:32:24.317 -> Address = 0x28 0xF5 0xC3 0x7 0xD6 0x1 0x3C 0x7C
12:32:24.350 ->
```

Figura 7. Monitor serie con las identificaciones de los sensores.

CÓDIGO PARA LEER LA TEMPERATURA DEL SENSOR

En este código se utilizan las librerías que nos permite leer la temperatura en grados celcius y además se direccionan tres posiciones que permiten leer las temperaturas de los tres sensores conectados.

Si solo necesitas un sensor eliminas el código donde se cambia el Index a 1 y 2 y las otras líneas de impresión.


```

// Skecht para leer datos de temperatura del sensor DS18B20, termocupla sumergible
digital
// Que utiliza protocolo One-wire, resolución configuragle de 9 a 12 bits
#include <DallasTemperature.h>
#include <OneWire.h>
OneWire ourWire(2); //Se establece el pin 2 como bus OneWire
DallasTemperature sensors(&ourWire); //Se declara una variable u objeto para nuestro
sensor
int Address=0;
void setup() {
delay(1000);
Serial.begin(9600);
sensors.begin(); //Se inicia el sensor
sensors.setResolution(Address, 12); // resolución de 9 bits
}
void loop() {
sensors.requestTemperatures(); //Se envía el comando para leer la temperatura
float temp= sensors.getTempCByIndex(0); //Se obtiene la temperatura en ºC la posición
cero
Serial.print("Temperatura T1= ");
Serial.print(temp);
Serial.println(" C");
sensors.requestTemperatures(); //Se envía el comando para leer la temperatura

temp= sensors.getTempCByIndex(1); //Se obtiene la temperatura en ºC la posición uno
Serial.print("Temperatura T2= ");
Serial.print(temp);
Serial.println(" C");
sensors.requestTemperatures(); //Se envía el comando para leer la temperatura

temp= sensors.getTempCByIndex(2); //Se obtiene la temperatura en ºC la posición dos
Serial.print("Temperatura T3= ");
Serial.print(temp);
Serial.println(" C");
delay(2000);
}

```

```
11 sensors.begin(); //Se inicia el sensor
12 sensors.setResolution(Address, 12); // resolución de 9 bits
13 }
14 void loop() {

Salida Monitor Serie x

Mensaje (Intro para mandar el mensaje de 'Arduino Uno' a 'COM4')

13:05:43.972 -> Temperatura T1= 39.13 C
13:05:44.520 -> Temperatura T2= 38.38 C
13:05:45.056 -> Temperatura T3= 13.88 C
13:05:47.584 -> Temperatura T1= 39.13 C
13:05:48.116 -> Temperatura T2= 38.38 C
13:05:48.688 -> Temperatura T3= 13.75 C
```

Figura 8. Monitor serie con las lecturas de los tres sensores DS18B20

REFERENCIAS BIBLIOGRÁFICAS

- [1] MAXIM, "DS18B20 Programmable Resolution 1-Wire Digital Thermometer," 2008. Accessed: Oct. 14, 2023. [Online]. Available: <https://pdf1.alldatasheet.es/datasheet-pdf/download/433921/MAXIM/DS18B20.html>

WEBGRAFIA

<https://programarfacil.com/blog/arduino-blog/ds18b20-sensor-temperatura-arduino/>
<https://electronilab.co/tienda/sensor-de-temperatura-ds18b20-tipo-sonda/>
<https://ferretronica.com/products/sensor-temperatura-termocupla-sumergible-ds18b20>
<https://www.sigmaelectronica.net/producto/sen-ds18b20-1m/>
<https://www.sigmaelectronica.net/wp-content/uploads/2018/09/DS18B20.pdf>
<https://cdn-shop.adafruit.com/datasheets/DS18B20.pdf>
https://github.com/Yese25/SENSOR_TEMP_DS18B20.git